

Object Names:

Database objects are composed of a four to twenty character identifier of mixed case followed by an underscore, a single character object type identifier (upper case), and finally a single digit numeric used to differentiate similar objects with the same identifier. An example of this would be "JobTransactions_T1" which indicates the first regular table of the series JobTransactions.

The single character object type identifiers are as follows:

Object Type	Code
Regular Table	T
Temporary ¹ Tables	TT
Persistent ² Temporary Tables	P
Development ³ Temporary Tables	D
Views	V
Stored Procedures	S
Rules	R
Defaults	D
Triggers	X
User Defined Datatypes	U
Indices/Unique Constraints	I
Check Constraints	K

Fields:

Field names are now composed of a four to twenty character identifier of mixed case, followed by an underscore, followed by a three to four character data type indicator in all uppercase letters (see list of approved codes at the end of this document.) For example: FirstName_TEXT.

Additionally, field names should always differ by more than simply the data type suffix. An example of the incorrect behavior would be if two fields were named Service_CODE and Service_IDNO. Instead, create distinct names which will help prevent using the wrong field during programming. A correct example would be ClientService_IDNO and ServiceType_CODE.

¹ Temporary Tables are tables where the structure is created at runtime by the application, the data contained in those tables is used only in temporary contexts and then the Table is destroyed. Examples would be extract tables where varying data may be stored and used during a runtime session but does not need to be retained at the end of the session.

² Persistent Temporary Tables are tables where the structure is permanent but the data contained in those tables is used only in temporary contexts. Examples would be storage tables where processing may store data during calculations but which may be deleted at any time.

³ Development Temporary Tables are tables used by developers during their programming. The identifier for these tables should contain a date in the format YYYYMMDD after which the table may be safely deleted.

Notes: When working in the mainframe environment "mixed-case" is not available, therefore substitute all single case. New applications should always use the standards outlined in this document, existing applications may choose one of three options. 1) Continue with current naming scheme. 2) Use a "hybrid" scheme of the node number + the new naming scheme (applies only to mainframe applications.) 3) Convert to total usage of the new naming scheme.

CLASS WORD	ABBREVIATION	DEFINITION
ADDRESS	ADDR	STANDARD MAILING ADDRESS DATA
AMOUNT	AMNT	ANY QUANTITY OF MONEY.
CODE	CODE	DATA WHICH IDENTIFIES CLASSIFICATIONS OR GROUPS OF ENTITIES.
DATE	DATE	ACTUAL CALENDAR DAY OR GROUP OF DAYS (I.E. MONTH AND YEAR).
DATE-TIME	DTTM	ACTUAL CALENDAR DAY OR GROUP OF DAYS (I.E. MONTH AND YEAR) PLUS ANY UNIT OF TIME NOT CONSIDERED A DATE.
IDENTIFICATION NUMBER	IDNO	NUMERIC OR ALPHANUMERIC DATA. IDENTIFIES A SPECIFIC ENTITY.
INDICATOR	INDC	BINARY OR ALPHANUMERIC VALUE (I.E.Y/N)
KEY	KEY	GROUP OF DATA USED AS A KEY FIELD
LOCATION	LCTN	GEOGRAPHIC POSITIONING DATA.
NAME	NAME	ALPHANUMERIC DATA THAT IDENTIFIES A SPECIFIC ENTITY.
NUMBER	NUMB	NUMERIC DATA THAT IDENTIFIES A SPECIFIC ENTITY.
PERCENT	PCT	RATIO BETWEEN DATA VALUES EXPRESSED AS A PERCENTAGE.
QUANTITY	QNTY	AN AMOUNT OF ANYTHING OTHER THAN MONEY.
TEXT	TEXT	FREE FORM CHARACTER DATA
TIME	TIME	ANY UNIT OF TIME OTHER THAN DATE
INTEGER	INT	ANY INTEGER NUMBER
IMAGE	IMG	IMAGE BINARY DATA (GIF, JPG, ETC)
USER DEFINED DATA	UDD	ANY PROGRAMMER DEFINED DATA TYPE
EMAIL ADDRESS	EML	TEXT REPRESENTATION OF EMAIL ADDR
URL	URL	INTERNET ADDRESS
BIT	BIT	NUMERIC BINARY (I.E., 1 or 0)

03/10/2008

SQL Abbreviation Usage for specific DB2 and SQLServer data types:

ABBREVIATION	DB2	SQLServer
ADDR	varchar	varchar
AMNT	decimal/float	decimal/float
Date	date	
TIME	time	
Date-Time	timestamp	datetime
IDNO	char/varchar/int	char/varchar/int
INDC	char(1)	char(1)/bit
NAME	varchar	varchar
NUMB	int	int
PCT		float
QNTY	int	int
TEXT	varchar	varchar
INT	int	int
IMG	vargraphic/dbclob	
EML	varchar	varchar
URL	varchar	varchar
BIT	char(1)	bit

Additional Standards:

- Varchar's should be a length of five or more.
- Email Address fields should be a varchar(35) or more.
- Audit Trails are strongly recommended.